A Brief History of Ajax

New technology quickly becomes so pervasive that it's sometimes hard to remember what things were like before it. The latest example of this in miniature is the technique known as Ajax, which has become so widespread that it's often thought that the technique has been around practically forever.

In some ways it has. During the first big stretch of browser innovation, Netscape added a feature known as LiveScript, which allowed people to put small scripts in web pages so that they could continue to do things after you'd downloaded them. One early example was the Netscape form system, which would tell you if you'd entered an invalid value for a field as soon as you entered it, instead of after you tried to submit the form to the server.

LiveScript became JavaScript and grew more powerful, leading to a technique known as Dynamic HTML, which was typically used to make things fly around the screen and change around in response to user input. Doing anything serious with Dynamic HTML was painful, however, because all the major browsers implemented its pieces slightly differently.

Shortly before web development died out, in early versions of Mozilla, Netscape showed a new kind of technique. I don't think it ever had a name, but we could call it Dynamic XML. The most vivid example I remember seeing was a mockup of an Amazon.com search result. The webpage looked just like a typical Amazon.com search result page, but instead of being written in HTML it was a piece of XML data which was then rendered for the user by a piece of JavaScript. The cool part was that this meant the rendering could be changed on the fly — there were a bunch of buttons that would allow you to sort the books in different ways and have them display using different schemes.

Shortly thereafter the bubble burst and web development crashed. Not, however, before Microsoft added a little-known function call named XMLHttpRequest to IE5. Mozilla quickly followed suit and, while nobody I know used it, the function stayed there, just waiting to be taken advantage of.

XMLHttpRequest allowed the JavaScript inside web pages to do something they could never really do before: get more data.[1] Before, all the data either had to be sent with the web page. If you wanted more data or new data, you had to grab another web page. The JavaScript inside web pages couldn't talk to the outside world. XMLHttpRequest changed that, allowing web pages to get more data from the server whenever they pleased.

Google was apparently the first to realize what a sea change this was. With Gmail and Google Maps, they built applications that took advantage of this to provide a user interface that was much more like a web application. (The startup Oddpost, bought by Yahoo, actually predated this but their software was for-pay and so they didn't receive as much attention.)

With Gmail, for example, the application is continually asking the server if there's new email. If there is, then it live updates the page, it doesn't make you download a new one. And Google Maps lets you drag a map around and, as you do so, automatically downloads the parts of it you want to look at inline, without making you wait for a whole new page to download.

Jesse James Garrett of Adaptive Path described this new tactic as Ajax (Asynchronous Javascript And XML) in an essay and the term immediately took off. Everyone began using the technique in their own software and JavaScript toolkits sprung up to make doing so even easier.

And the rest is future history.

Both systems were relatively ill-supported by browsers in my experience. They were, after all, hacks. So while they both seemed extremely cool (KnowNow, in particular, had an awesome demo that allowed for a WYSIWYG SubEthaEdit-style live collaboration session in a browser), they never really took off.

Now apparently there is another technique, which I was unaware of, that involved changing the URL of an iframe to load new JavaScript. I'm not sure why this technique didn't quite take off. While Google Maps apparently used it (and Oddpost probably did as well), I don't know of any other major users.

As my commenters point out — and as I well knew, but momentarily forgotten — this isn't really true. Before XMLHttpRequest, people used a trick of not closing the connection to the server. The server would keep adding more and more to the page, never saying it had finished downloading. Ka-Ping Yee used this technique to make a real-time chat system based on an animated GIF. And the ill-fated startup KnowNow used a similar technique with JavaScript to allow for live-updating pages. ■

You should follow me on twitter here.

December 22, 2005